



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/815,018	03/26/2004	Gregor K. Frey	6570P033	1010
8791 7590 09/03/2008 BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP 1279 OAKMEAD PARKWAY SUNNYVALE, CA 94085-4040				
EXAMINER VU, TUAN A				
ART UNIT 2193		PAPER NUMBER		
MAIL DATE 09/03/2008		DELIVERY MODE PAPER		

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

# Office Action Summary

Application No.

10/815,018

Applicant(s)

FREY ET AL.

Examiner

TUAN A. VU

Art Unit

2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

## Status

- 1) ☒ Responsive to communication(s) filed on 5/21/08.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

## Disposition of Claims

- 4) ☒ Claim(s) 1-28 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-28 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

## Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

## Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
  - ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

## Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO/SF/ICE)  
Paper No(s)/Mail Date 5/21/08 1: 5/21/08 2
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: \_\_\_\_\_

### DETAILED ACTION

1. This action is responsive to the Applicant's response filed 5/21/08.

As indicated in Applicant's response, claims 1, 3, 10, 16, 20, 23 have been amended.

Claims 1-28 are pending in the office action.

#### *Double Patenting*

2. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

3. Claim 1 is provisionally rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claim 15, 24 of copending Application No. 10,749,616 (hereinafter '616). Although the conflicting claims are not identical, they are not patentably distinct from each other because of the following observations.

**As per instant claim 1**, '616 recites 'tracing module ... to receive and process method calls by the application when the specified regions are executed' and 'logging module associated with specified categories of the system ... to receive and process method calls from components

associated with the categories' in a system of 'computers interconnected through a network'; a log message *formatters* to convert trace or log method calls to specified message formats (i.e. a obvious variant of 'determine message format for the received message'). '616 claim 15 does not recite formatter including' a configuration file storing a format' which is destined for the tracing module or logging module, the formatter indicating thereby a format for a message to be sent from said modules 'according the indicated format'; nor does '616 claim 15 recite 'output destination to receive the formatted message'. However, for one of ordinary skill in the art interpreting the message formatters by '616 from above, this 'output destination' and 'configuration file storing a format' limitation would have been an obvious features by which '616 invention would necessarily implement a format indication to be received at the log/trace modules such that this indication about proper format (regarding '616 messages to be effectuated) would be necessary for such properly formatted message to get to their destination by an output module.

Likewise, '616 claim 24 for reciting the limitations of '616 claim 15 is also an obvious variation of instant claim 1.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

4. Claim 1 is provisionally rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claims 6, 17, 23, 30 of copending Application No. 10,813,999 (hereinafter '999). Although the conflicting claims are not identical, they are not patentably distinct from each other because of the following observations.

**As per instant claim 1**, '999 claim 6 recites 'tracing module ... specified program regions ... to receive and process method calls by the application when the specified regions are executed' and 'logging module associated with specified categories of the system ... to receive and process method calls from network components associated with the categories'; a user interface to configure an output destination via a dialog window(for the tracing and logging system). Although '999 does not recite formatter including' a configuration file storing a format' which is destined for the tracing module or logging module, the formatter indicating thereby a format for a message to be sent from said modules 'according the indicated format'; nor does '999 recite 'output destination to receive the formatted message' from logging and tracing module and a 'formatter' to determine message format therefor. However, for one of ordinary skill in the art, the concept of 'output destination' and formatting is suggestive of a message being received in '999 integrated system, and '999 use of GUI for setting attributes suggests receiving of a form of configuration in order to sort out attributes related to compliant format for messages to be sent out. One of ordinary skill in the art would have construed '999 GUI for setting an attribute of such output destination as a obvious language variant to the instant claim's recital of 'formatter' and configuration file indication of a format needed for '999 logging module to effectuate properly format in said outgoing messages.

Likewise, '999 claims 17, 23, 30 recite a tracing controller, a logging controller, an output destination, and a GUI dialog for setting attributes for said output destination, all of which being similar to '999 claim 6; hence are deemed obvious variations of instant claim 1.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

5. Instant claim 16 recites subject matter of claim 1, and should be rejected as incorporating the same type of obviousness rationale from above.

***Claim Rejections - 35 USC § 101***

6. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

7. Claims 1-9, 16-19 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

The Federal Circuit has recently applied the practical application test in determining whether the claimed subject matter is statutory under 35 U.S.C. § 101. The practical application test requires that a “useful, concrete, and tangible result” be accomplished. An “abstract idea” when practically applied is eligible for a patent. As a consequence, an invention, which is eligible for patenting under 35 U.S.C. § 101, is in the “useful arts” when it is a machine, manufacture, process or composition of matter, which produces a concrete, tangible, and useful result. The test for practical application is thus to determine whether the claimed invention produces a “useful, concrete and tangible result”.

The current focus of the Patent Office in regard to statutory inventions under 35 U.S.C. § 101 for method claims and claims that recite a judicial exception (software) is that the claimed invention recite a practical application. Practical application can be provided by a physical transformation or a useful, concrete and tangible result. The following link on the World Wide Web is the United States Patent And Trademark Office (USPTO) reference in terms of guidelines on a proper analysis on 35 U.S.C. §101 rejection.

[http://www.uspto.gov/web/offices/pac/dapp/opla/prconnotice/guidelines101\\_20051026.pdf](http://www.uspto.gov/web/offices/pac/dapp/opla/prconnotice/guidelines101_20051026.pdf)

Specifically, claim 1 recites ‘an integrated ... system’ comprising a tracing module, a logging module, a output destination and a formatter. According to the Specifications (e.g. Drawings: Fig. 2-4), these modules and formatter amount to software-based entities and programmatic functionality. As a whole, the system cannot be construed as having hardware

support to execute what appears to be mere software functionality. Therefore, the claim for reciting mere 'Functional Descriptive Material' (see USC101 Guidelines, Annex IV, pg. 52-54) not only fails to qualify as being one of the 4 statutory categories of subject matter (emphasis added), but also cannot fulfill the requirement of practical application; that is, not able to yield data transformation with concrete, useful, and tangible result.

Claim 1 and dependent claims 2-8 are therefore rejected for non-statutory subject matter.

Claim 16 recites system with means for creating tracing/logging controller, for specifying output destination, and a formatter. The recited tracing/logging 'controller' and means for creating instance thereof, and 'formatter' are construed as software entities without being embodied in any hardware medium or tangible support enabling the realization of their respective functions. The means to create those software entities is not conveyed from the Disclosure (e.g. Specs: Figure 9) as being hardware means, but rather as software means effectuated within an computer application. As set forth above, the claim for reciting mere 'Functional Descriptive Material' cannot be categorized as statutory subject matter, nor can it fulfill the requirement of practical application; that is, not able to yield data transformation with concrete, useful, and tangible result. Claim 16 and dependent claims 17-19 are therefore rejected for non-statutory subject matter.

### ***Claim Rejections - 35 USC § 103***

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

9. Claims 1-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over James Hart, "Early Adopter: J2SE 1.4", chapter 5, September 2001, Wrox Press, pp. 1-12 (hereinafter Hart – refer to 'EarlyAdopter\_DS.pdf' in PTO -892) in view of APA (Admitted Prior Art: see Specifications pg. 4, para 0008-0009).

**As per claim 1**, Hart discloses an integrated tracing and logging system employed within a network comprising:

a tracing methods (e.g. *methods ... can be used ... debugging trace of program activity* – pg. 6, bottom half; *part of this API ... throws IllegalArgumentException* – pg. 7, top half) to receive via an application programming interface (the Logging API, pg. 2; void Methods – pg. 5 reads on APIs ) and process tracing method calls generated by the application;

a logging module associated with specified categories related to the network (e.g. *specified namespace, hierarchy naming*- pg. 4, bottom; void *setLevel* – pg. 4 – Note: manager to set level for each level of hierarchy for spawning one logger reads on specified categories of the network – e.g. *setLevel()*, pg. 9, bottom), the logging module to receive via the API and process logging method calls (e.g. Logging Methods - pg. 5-7) from network components associated with the categories (*distributed applications... number of networked machines*, pg. 1 );

a formatter to provide an indication of the format to the tracing/logging module via the API (formatters – 2<sup>nd</sup> para , pg. 5); wherein processing tracing/logging method calls includes receiving from the formatter an indication of a format for a message (e.g. string message ... formatters – pg. 5, 2<sup>nd</sup> para) to be sent from the respective tracing/logging module, the respective tracing/logging module further to format the message according to the indicated format (what order ... what parameters – bottom half, pg. 6);



an output destination (e.g. *different destination; log files, the console, ... in-memory buffer* – pg. 2, top para; *fired off ... different types of storage* – pg. 1) to receive the formatted message (*Logging API... Loggers: pass messages for logging* – pg. 2; *Messages are passed* – pg. 2 bottom) from at least one of the tracing module and the logging module.

But Hart does not explicitly disclose formatter including a configuration file storing a format, so that the formatter can indicate a format to the tracing module. Based on Hart teaching of a parameter record to be passed as input for XML formatting (e.g. *record passed to a Formatter* – bottom, pg. 3; *record parameters ... can be used by formatters* - 2<sup>nd</sup> para, pg. 5; *verbose XML formatted data ... LogRecord's parameters* – bottom pg. 10), it would have been obvious for one skill in the art at the time the invention was made to implement the formatters by Hart so that they incorporate a parameters record as mentioned above, and use this as an input file containing format/configuration based on which the formatters provide more information (such as condition of an exception, warning) to the loggers other than formatting a text alone, as suggested in Hart (*more information ... error condition ... useful information* – 2<sup>nd</sup> para pg. 5; *level.SEVERE, level.WARNING* – pg. 3)

Hart does not explicitly disclose *tracing module associated with specified program code regions of an application* to receive and process tracing method calls generated by the application when the specified program code regions are executed. APA teaches developers using logging techniques in tight conjunction with tracing of executing program code, and tracing tool being equally used with logging also require sending messages to console or other output destination. Based on the API for effectuating calls related to debug (see Hart: *methods ... can be used ... debugging trace of program activity* – pg. 6, bottom half;) of a particular area

covered by the logger (see Hart, bundleName, Object params - pg. 5-7 – Note: a instantiated logger class to analyze *Object params* of *bundle Name* reads on logger class for a particular name instance, or region ), it would have been obvious for one skill in the art at the time the invention was made to implement Hart method with a tracing API/module specific for a particular regions – or namespace within a hierarchy -- corresponding the specific instance of *logger* among many other concurrent logger classes being created in order for such dedicated trace module to monitor method calls generated by the application when the specified program code regions are executed, and this would be consistent with the endeavor by developers to attach a debug module with a logging module as set forth by APA, and in view of Hart's purport to provide both debug/tracing information into a dedicated logger being named for a specific naming instance.

**As per claim 2**, Hart discloses a markup language formatter (e.g. filters and formatters ... information about logging event – pg. 5, 2<sup>nd</sup> para; *XML file, level threshold set to INFO ... precise configuration ... Java properties format* - pg. 7, bottom; bottom half pg. 8)

**As per claims 3-4, and 6**, Hart discloses wherein one or more properties (e.g. log.severe – pg. 8, top; *more information ... it also contain an exception* – 2<sup>nd</sup> para, pg. 5) of the formatter are defined in the configuration file; wherein the configuration file includes an identifier (e.g. *XML file, level threshold set to INFO ... precise configuration ... Java properties format* - pg. 7, bottom; *public class Logging* – pg. 8; `<class>Logging</class>` - pg. 8, bottom; *LogRecord's parameters* – bottom pg. 10) to identify the formatter (Note: logging class with Java handlers using INFO set from the XML reads on formatter); wherein the configuration file defines the

message format for the received message, the message format including one or more fields (e.g.

*XML details ... <record> ... </record>* pg. 8-9).

**As per claim 5**, Hart discloses wherein the one or more properties are formatted as key-value-pair properties, each key-value pair having a key to specify an attribute (<date> ... </date>, <logger> ... </logger> pg. 8) and a value to provide a definition (e.g. *998524070390.com.wrox.ea.j2se.utilities.Logging* – pg. 8) for the specified attribute.

**As per claim 7**, Hart discloses wherein the one or more fields of the message format includes

at least one of a timestamp field (e.g. *date, millis* – pg. 8, bottom – Note: record storing date and millis from a XML used for firing message reads on time of received message) to indicate a time for the received message;

a location of origin field to indicate a source (String sourceClass – pg. 5; <method>, pg. 8) of the received message;

a thread identifier field to indicate a thread (<thread> - pg. 8, bottom) associated with the received message;

a message severity indicator field to indicate a severity level (Level level – pg. 5; warning – pg. 8 bottom) of the received message; and

a message identifier field to identify the received message (String msg – pg. 5; <message> pg. 8, bottom).

**As per claims 8-9**, Hart discloses wherein the output destination is at least one of a trace file; and a log file, a console (e.g. *different destination; log files, the console ... in-memory buffer* – pg. 2, top para).

**As per claim 10**, Hart discloses a computer-implemented method employed within a network comprising creating an instance of:

a tracing object (refer to tracing methods by Hart in claim 1) to receive and process tracing method calls generated by the application;

a logging controller associated with specified categories related to the network, the logging controller to receive and process logging method calls from network components associated with the categories (*distributed applications... number of networked machines*, pg. 1 ) (re claim 1);

providing a common application programming interface of the tracing controller instance and logging controller instance, whereby the tracing and logging controller instances are accessed (*the Logging API: ... framework of classes* pg. 2, top)

specifying an output destination to receive via the common application programming interface of the tracing controller instance and logging controller instance (see pg. 2) a message from at least one of the tracing controller instance and the logging controller instance (re claim 1); and

selecting a formatter (e.g. *Logging Messages*, pg. 5-7 - Note: specifying a level -- or filtering -- by the main *Logger* object in order to instantiate a localized record logger reads on selecting a level-bound class to perform logging using its format handlers, i.e. formatter being selected by the filtering and naming process) to provide a message format for the received message; wherein the message format is defined based, at least in part, on a configuration file (refer to claim 3).

Hart does not explicitly disclose instance of a *tracing controller associated with specified program code regions of an application* (to receive and process), *when the specified program code regions are executed*. However, the tracing controller would falls under the ambit of addressing the 'tracing module' for 'specified code regions' in claim 1; hence will be rejected herein including the rationale as set forth therein.

**As per claims 11-12**, refer to claims 2, 4 (Note: selected formatter functions to configure message using the XML in claim 2 reads on configuring for the selected formatter).

**As per claims 13-14**, refer to claims 6-7.

**As per claim 15**, Hart discloses a filter to the specified output destination to selectively filter the message.

**As per claim 16**, Hart discloses a system comprising

a means for creating

an instance of a tracing object to receive via a application programming interface ( Logging API, pg. 2) and process tracing method calls generated by the application (refer to claim 1);

an instance of a logging controller associated with specified categories related to the network, the logging controller to receive via a application programming interface ( Logging API, pg. 2) and process logging method calls from network components associated with the categories (*distributed applications... number of networked machines*, pg. 1 );

a formatter to provide an indication of the format to the tracing/logging module via the API (formatters – 2<sup>nd</sup> para , pg. 5); wherein processing tracing/logging method calls includes receiving from the formatter an indication of a format for a message (e.g. string message ...

Art Unit: 2193

formatters – pg. 5, 2<sup>nd</sup> para) to be sent from the respective tracing/logging module, the respective tracing/logging module further to format the message according to the indicated format (what order ... what parameters – bottom half, pg. 6);

a means for specifying an output destination to receive the formatted message from the respective tracing controller instance or the logging controller instance;

all of which having been addressed in claim 10.

But Hart does not explicitly disclose formatter including a configuration file storing a format, so that the formatter can indicate a format to the tracing module. But this limitation has been rendered obvious as set forth in claim 1.

Hart does not explicitly disclose instance of a *tracing controller associated with specified program code regions of an application* (to receive and process), *when the specified program code regions are executed*. However, the tracing controller would falls under the ambit of addressing the ‘tracing module’ for ‘specified code regions’ in claim 1; hence will be rejected herein including the rationale as set forth therein.

**As per claims 17-18**, refer to claims 12-13.

**As per claim 19**, refer to claim 14.

**As per claim 20**, Hart discloses article of manufacture comprising an electronically accessible medium providing instructions that, when executed by an apparatus cause the apparatus

to create an instance of

a tracing object to receive and process tracing method calls generated by the application (refer to claim 10);

a logging controller associated with specified categories related to the network, the logging controller to receive and process logging method calls from network components associated with the categories (refer to claim 10);

to provide a common application programming interface of the tracing controller instance and logging controller instance, whereby the tracing and logging controller instances are accessed (refer to claim 10)

to specify an output destination to receive via common application programming interface of the tracing controller instance and logging controller instance (see above) a message from at least one of the tracing controller instance and the logging controller instance; and select a formatter (refer to claim 10) to provide a message format for the received message, wherein the message format is defined based, at least in part, on a configuration file;

all of which having been addressed in claim 10.

Hart does not explicitly disclose instance of a *tracing controller associated with specified program code regions of an application* (to receive and process), *when the specified program code regions are executed*. However, this tracing controller (to operate on a specified code regions) has been addressed using the rationale as set forth in claim 1.

**As per claims 21-22**, refer to claims 12-13.

**As per claim 23**, Hart discloses an apparatus comprising: an application; and a processor and logic executable thereon to create and specify the same elements as recited in claim 20, hence would integrate the corresponding rejection as set forth therein, including the rationale as to render obvious the ‘tracing controller’ limitation recited as *associated with specified program*

*code regions of an application (to receive and process), when the specified program code regions are executed.*

**As per claims 24-25**, refer to claims 2-3.

**As per claims 26-27**, refer to claims 11, 13.

**As per claim 28**, refer to claim 14.

***Response to Arguments***

10. Applicant's arguments filed 5/21/08 have been fully considered but they are moot and/or not persuasive. Following are the Examiner's observation in regard thereto.

**USC § 101 Rejection:**

(A) Applicants have submitted that 'configuration file' and message format contribute to the statutory requirement for a hardware support (Appl. Rmrks, pg. 9). The amendment is deemed insufficient in providing hardware to realize functionality of the very software functionality (controller modules, formatter) using this configuration file or message. The rejection will be maintained, because the claimed subject matter amounts to listing of software elements only (refer to claim rejection); which furthermore, cannot be categorized as article of manufacturer, apparatus, composition of matter or process steps.

**USC § 103 Rejection:**

(B) Applicants have submitted that claims 1, 10, 16, 20 and 23 as amended include 'application programming interface'; and that Hart does not teach or suggest a common API of the tracing/logging module. This added limitation has been addressed with corresponding cited portions as set forth in the rejection responsive to the added language. The arguments for not being commensurate with a previously submitted ground of rejection would be considered **moot**



in view of the current and latest state of the rejection, which has been necessitated by the above Amendments.

In all, the claims stand rejected as set forth in the Office Action.

***Conclusion***

11. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (571) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lewis Bullock can be reached on (571)272-3759.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 ( for non-official correspondence - please consult Examiner before

Art Unit: 2193

using) or 571-273-8300 ( for official correspondence) or redirected to customer service at 571-272-3609.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Tuan A Vu/

Primary Examiner, Art Unit 2193

August 19, 2008